UpGuard

The Email Security Checklist

Overview

You've hardened your servers, locked down your website and are ready to take on the internet. But all your hard work was in vain, because someone fell for a phishing email and wired money to a scammer, while another user inadvertently downloaded and installed malware from an email link that opened a backdoor into the network. Email is as important as the website when it comes to security. As a channel for social engineering, malware delivery and resource exploitation, a combination of best practices and user education should be enacted to reduce the risk of an email-related compromise.

By following this 13 step checklist, you can make your email configuration resilient to the most common attacks and make sure it stays that way.

1. Enable SPF

How do you know if an email is really from who it says it's from? There are a couple of ways to answer this question, and Sender Policy Framework (SPF) is one. SPF works by publishing a DNS record of which servers are allowed to send email from a specific domain.

- An SPF enabled email server receives an email from somebody@example.com
- The email server looks up example.com and reads the SPF TXT record in DNS.
- If the originating server of the email matches one of the allowed servers in the SPF record, the message is accepted.

SPF should be enabled on all edge email systems to ensure that both emails coming into your organization can be checked for SPF and that emails coming from your organization can't be impersonated by someone using an email server not listed in the SPF record. Failure to use SPF means your emails can't be checked for an authentic origination server and are therefore far less trustworthy than those that can.

2. Enable DKIM

Unlike SPF, which applies on a per-domain basis, DomainKeys Identified Mail (DKIM) adds an encrypted signature on every message that can be validated by a remote server against a DNS TXT record.

Essentially, organizations claim responsibility for email messages with their DKIM signature. Because signatures are encrypted, they are very difficult to forge; therefore an organization's reputation is on the line for messages sent out under their DKIM signature. DKIM signatures will be ignored by mail servers that do not support it, so there's no worrying about whether all your recipients are DKIM compatible. Failure to use DKIM reduces the integrity of your email and increases the likelihood of your domain being blacklisted.

3. Enable DMARC

The Domain-based Message Authentication, Reporting and Conformance (DMARC) protocol builds on SPF and DKIM to handle verification of sender domains. DMARC also provides reporting to give organizations visibility into their email policy. Additionally, DMARC specifies what to do with a message if the SPF and DKIM authentication mechanisms fail. The combination of SPF, DKIM and DMARC creates a trustworthy email environment. All three rely on DNS TXT records to work, so be sure to see step 11 on securing DNS. Failure to use DMARC means that SPF and DKIM policies will be different depending on where the message is sent. DMARC standardizes that by including instructions within the email itself.

4. Set up a Spam Filter

This one probably seems straightforward: you need some mechanism to block spam. Around 50% of all total email is considered spam. This number is actually falling as botnet defense has improved. However, a spam filter is still crucial to any organization. A dedicated appliance such as what Barracuda offers can handle large quantities of email and easily fits in front of your edge servers. Alternatively, a cloud-based spam filter allows you to offload the duties to a provider. Whichever spam filter you choose, be sure to do the following:

- Subscribe to a DNS blackhole list. This will block most of the spam right at the edge.
- Enable rate control to prevent remote senders from overwhelming the server.
- Enable content analysis to heuristically block or quarantine probable spam.
- Block senders who fail reverse DNS lookups.
- Filter dangerous attachments. (See step 8 of this guide).

There are other ways to protect against spam, but configuring these 5 on a dedicated spam filter should clean up your mail delivery and have a minimal number of false positives.

Failure to set up a spam filter will mean most of the email your organization processes is garbage and people's inboxes will be near impossible to use.

5. Disable Relaying

Relaying is the process of connecting to an email server and sending mail through that server to a third party. Here's an example of how this can go terribly wrong:

- A spammer connects to a random email server with open relay accidentally configured.
- The spammer sends thousands (tens of thousands, hundreds of thousands) of emails through that server to various email providers across the internet.
- The server the spammer connected to is then blacklisted for spam, blocked by several major email carriers such as Yahoo and Gmail, which prevents legitimate messages from that server getting through.
- The spammer searches for another open relay server to use, while the team responsible for the open relay server has to fight to get themselves off the blacklists.

Typically, relaying should always be disabled. When it isn't disabled, it should only be allowed for specific IP addresses of servers you manage or trust completely.

Failure to do this can get you blacklisted, as in the scenario above, and otherwise overload your email servers and prevent normal mail delivery to and from your domain.

6. Set a Throttling Policy

In step 5, we saw how an open relay can allow a spammer to send massive amounts of email through a misconfigured server. But sometimes a legitimate user becomes a spammer because they fell for a phishing scam or otherwise had their password compromised, and their account is now being used by someone else to send spam. In either case, setting a reasonable throttling policy will prevent spammers from sending the amount of email necessary to get your domain blacklisted.

The actual numbers will vary from environment to environment, but it's best to get an idea of how much legitimate email goes through your system each day and build a policy around that. If you send legitimate bulk emails such as marketing materials, consider utilizing a remailer service such as SendGrid. Not only will that greatly reduce your chances of getting blacklisted, you'll likely have access to important metrics and reports on your email delivery as well.

Consider throttling these things:

- Number of recipients per sender per day.
- Number of emails per sender per hour.
- Number of recipients per emailBlock senders who fail reverse DNS lookups.

There are other ways to throttle email, but by restricting the above actions to just above what your company actually uses, you can prevent a compromised account or server from damaging your domain's reputation enough to blacklist you.

7. Restrict Local Email Domain

Occasionally you'll see phishing emails that seem to come from your own domain, but originated on an internet mail server somewhere. To prevent this, you can restrict emails that come from your domain to only be allowed from your mail system. This means that any emails coming from the internet claiming to be from your domain will be blocked. This increases security because people are more likely to fall for a phishing scam that claims to be from their own domain.

Before implementing this step, consider if there are any legitimate emails coming from the internet from your domain. Some examples of this would be:

- Using a remailer service (mail goes from your users, to the third party internet remailer, then back to your organization).
- Cloud services set to send as domain users.
- Web application forms that trigger emails sent from an internet web server with a domain email address.

Depending on your mail system, you may be able to allow email from specific hosts while still blocking it from the internet at large. Failure to do this can result in phishing emails sent to your users that seem to come from your email domain, which can easily trick someone into giving away their password or other important information.

8. Set Attachment Restrictions

Email attachments have long been an effective malware delivery system, so it's important to restrict the types of attachments that come through your server. The most dangerous file types are executables, so extensions such as .exe, .bat, .vbs, .jar, .swf and so forth should always be blocked. Those files can be transferred by other methods and/or zipped into an archive if they need to be shared for legitimate reasons. Better is to only allow file types you want, such as office documents, PDFs and pictures, depending on your business requirements. Failure to do this can lead in users inadvertently installing malware or otherwise have malicious attachments compromise their systems.

The edge spam filter mentioned in step 4 should be able to scan incoming and outgoing attachments for viruses. This will help protect against infected legitimate file types. Furthermore, encrypted or otherwise unscannable archives should be blocked as well. Password protected zip files might be nice, but if the virus scanner can't tell what it is, you shouldn't let it into your organization.

Attachment size should also be restricted. This too depends on your organization, but the average email attachment limit is rarely over 25MB. So even if your limit is higher, emails with large attachments will likely be blocked by the remote server anyway. There are many better ways to share files across the internet than email. Failure to limit attachment size can lead to huge user mailboxes, which will reduce performance and make migrations, backups and other operations less reliable for those mailboxes, especially in Microsoft Exchange environments.

9. Ensure Log Visibility and History

Whether you use Exchange, Postfix, or even a hosted email service, email management comes down to having and being able to make use of log files. Common administrative tasks like determining whether an email was delivered and complex server migrations both rely on quick and efficient use of logs.

- Create a log retention policy. This will be the period of time that you can be reasonably expected to go back and have log data. This should suit your business' needs, as well as meet any legal requirements for your industry. Failure to have this policy can cause chaos within an organization in the event of a disaster. Business expectations should be realistic and well documented.
- Make sure you have enough disk space for your logs. Systems administration 101, right? Yet many production servers are built to a spec that only accounts for the application resources and doesn't factor in tertiary needs such as logging, backup or monitoring. You need enough space to support the company's retention policy.
- Consider using log visualization tools that can efficiently summarize the huge amounts of data into a dashboard or other convenient format. But even if you do use one, be sure you know how to read the raw logs as well, it's better not to learn in an emergency.

Every piece of your mail environment (spam filters, edge servers, database servers, mtas) will likely have logs, so be sure to know what information is where and how to access it. Exchange has its own set of logs that should be monitored and troubleshot if necessary.

10. Consider Email Encryption

The only way to ensure end-to-end privacy for email is to encrypt the email itself between the sender and the recipient. Encrypted email allows a sender and receiver to exchange public keys for encryption, while retaining private keys for decryption, meaning that only the owner of the private key can access the contents of the email. There are several encryption options, including PGP and S/MIME.

Most modern email clients support certificate based email encryption, but be aware that unlike many of the other measures so far, encryption requires configuration for both the sender and the receiver. Certificates must be trusted by each party in order to work. Since encryption can reduce ease of use, only encrypt messages when the content warrants strict privacy.

11. Enable DNSSEC

Because so many crucial pieces of email security rely on DNS records, it's important to make sure that DNS itself is secured. DNS spoofing is a type of attack where a hacker reroutes a valid DNS address to the IP of a malicious server. For example:

- Someone goes to www.example.com in a browser.
- Their computer performs a DNS lookup for www.example.com.
- Example.com's DNS cache has been poisoned, so it returns the IP of a hacker's server.
- An imposter webpage from the hacker's server is loaded in the browser.
- User credentials are entered into the fake page, and stored in the hacker's database.

There are several other ways to exploit DNS, but DNSSEC prevents these by signing the DNS response using public key cryptography. This means that only authenticated DNS responses will be returned. When considering the above cases of SPF, DKIM and DMARC, it should be clear how protecting DNS records is essential to email security-- if the TXT records for these services are spoofed, it completely undermines the security they provide.

12. Educate your community

No matter what technological measures are put in place, ultimately security comes down to people and their day to day practices. A culture that actually cares about security and sees it as a business goal will have much better success implementing security technology than one that sees it only as an obstruction or cost.

Users should be educated on basic email security concepts, such as:

- How to avoid *ishing scams.
- Which files should be sent through email and what alternative file transfer methods are available.
- How to avoid malware and malicious links.
- How to detect social engineering and what information is okay to share.

This could be a basic course that's required as a condition of employment, perhaps refreshed on an interval to make sure practices keep pace with change. Without a program such as this, and support from upper management for a culture of security, none of the other measures mentioned here will matter much because careless user practice will eventually create backdoors into the network and/or a data breach.

13. Regularly Test Configurations

Visibility is the most important factor when it comes to hardening a server. Without knowing what is going on, what has changed and what needs to change, there's little hope of keeping a server secure over time. Regularly testing configurations against company policy will give IT teams a chance to fix security holes before they are exploited.

Furthermore, regular configuration testing pushes data centers towards standardizing their processes and streamlining workflows-- strong visualizations and historical trend data allow better and quicker decisions when it comes to making new changes. None of the other steps will make as much of an impact on security if they are not routinely tested.

Conclusion

Email security is a broad topic, but by following these 13 steps, you can be reasonably sure you have protected yourself against the most common attacks.

Reputation is tied to email, so taking steps to ensure only authorized communications come from your company's domain protects more than just their digital assets.

As a medium of social engineering, taking steps to educate people on phishing scams and other malicious email activity will nicely complement a well-configured spam filter to prevent data breaches and malware installation. Continuous testing of configuration ensures potentially dangerous changes are discovered and remediated in a timely manner.

Get a free snapshot of your organization's security performance.

We'll give you a quick view of your organization's website security performance across 13 risk factors, such as email security, SSL, DNS health and common vulnerabilities.

UpGuard

A Quick Email Security Checklist

- [ ] **1. Enable SPF**
  Prevent sender address forgery
- [ ] **2. Enable DKIM**
  Make your emails trustworthy
- [ ] **3. Enable DMARC**
  Utilize SPF and DKIM to the fullest
- [ ] **4. Set up a spam filter**
  Block spam before users see it
- [ ] **5. Disable Relaying**
  Prevent unauthorized use
- [ ] **6. Set a Throttling Policy**
  Prevent spamming and blacklisting
- [ ] **7. Restrict Local Email Domain**
  Prevent sender domain forgery
- [ ] **8. Set Attachment Restrictions**
  Prevent malicious attachments
- [ ] **9. Ensure Log Visibility and History**
  Know what's happening and what happened
- [ ] **10. Consider Email encryption**
  Guarantee privacy of email data
- [ ] **11. Enable DNSSEC**
  Prevent unauthorized DNS changes
- [ ] **12. Educate Your Community**
  Security starts with people
- [ ] **13. Regularly Test Configurations**
  Prevent configuration drift

UpGuard

Questions? We have answers

We're here to help, shoot us an email at sales@upguard.com

Know your vendors. Secure yourself.

Looking for a better, smarter way to protect your data and prevent breaches?

UpGuard offers a full suite of products for security, risk and vendor management teams.

Contact sales | Free demo →

Trusted by hundreds of companies worldwide

www.upguard.com

+1 888-882-3223

650 Castro Street, Suite 120-387, Mountain View CA 94041 United States